

LAPORAN RESMI
PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK
PENGANTAR CLASS DIAGRAM



Fadilah Fahrul Hardiansyah S.ST., M. Kom

Ratri Maria Manik

3121600039

D4 TEKNIK INFORMATIKA – B

PROGRAM STUDI TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
TA 2022/2023

A. PENDAHULUAN

1. Sebutkan bagian – bagian dari class diagram!

a. Class name

Yang merupakan sesuatu yang mewakili dari nama kelas. Nama kelas suatu diagram harus dapat menjelaskan atau mendeskripsikan maksud dari class tersebut (simple dan deskriptif)

Deklarasi untuk nama class pada suatu class diagram yaitu

```
<modifier> class <classname>
public class Mahasiswa{
.....
}
```

b. Atribut

Merupakan property atau object dari sebuah kelas yang digunakan saat mendeskripsikan instance tertentu dari class.

Deklarasi untuk atribut pada suatu class yaitu

```
<modifier> <type> <nama_atribut>
public int nrp;
public String nama;
```

c. Method

Merupakan sesuatu yang dapat dilakukan pada sebuah object dan berlaku untuk semua object yang terdapat dalam class tersebut.

Deklarasi untuk method pada suatu class yaitu

```
<modifier> <return_type> <nama_method> ([daftar argumen]){
    {<statement>}
}
public void getNrp(){
    return nrp;
}
```

2. Sebutkan relasi yang digunakan dalam class diagram!

a. Association

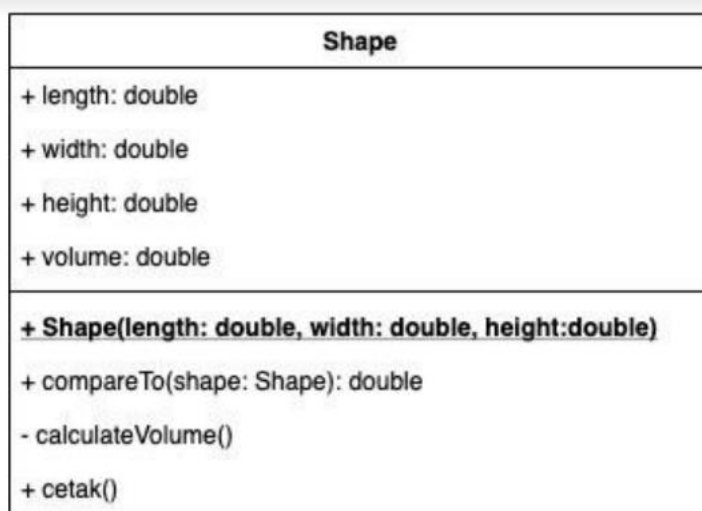
Suatu hubungan antara bagian dari dua kelas. Terjadi association antara dua kelas jika salah satu bagian dari kelas mengetahui yang lainnya dalam melakukan suatu kegiatan. Di dalam diagram, sebuah association adalah penghubung yang menghubungkan dua kelas.

b. Aggregation. Suatu association dimana salah satu kelasnya merupakan bagian dari suatu kumpulan. Aggregation memiliki titik pusat yang mencakup keseluruhan bagian. Sebagai contoh : OrderDetail merupakan kumpulan dari Order.

c. Generalization. Suatu hubungan turunan dengan mengasumsikan satu kelas merupakan suatu superClass dari kelas yang lain. Generalization memiliki tingkatan yang berpusat pada superClass. Contoh : Payment adalah superClass dari Cash, Check, dan Credit

B. LATIHAN

1. Mengubah class diagram menjadi kode program



- Listing Program Class Shape

```
public class Shape{
    public double length;
    public double width;
    public double height;
    public double volume;

    public Shape(double length, double width, double height){
        this.length = length;
        this.width = width;
        this.height = height;
    }

    public double compareTo(Shape shape){
        return (Math.abs(volume - (shape.length * shape.width * shape.height)));
    }

    public void calculateVolume(){
        volume = length * width * height;
    }

    public void cetak(){
        System.out.printf("panjang x lebar x tinggi = %.0f x %.0f x %.0f\n",
length, width, height );
        calculateVolume();
        System.out.println("Volume = " + volume);
    }
}
```

- Listing Program ShapeTester

```
public class ShapeTester{
    public static void main(String args[]){
        Shape kubus = new Shape(5, 5, 5);
        Shape balok = new Shape(10, 5, 5);
    }
}
```

```

        kubus.cetak();
        balok.cetak();
        Double selisihVolume = kubus.compareTo(balok);
        System.out.println("Selisih Volume = " + selisihVolume);
    }
}

```

Hasil Compile

```

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 6>javac ShapeTester.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 6>java ShapeTester.java
panjang x lebar x tinggi = 5 x 5 x 5
Volume = 125.0
panjang x lebar x tinggi = 10 x 5 x 5
Volume = 250.0
Selisih Volume = 125.0

```

C. TUGAS

Buatlah 3 buah class diagram lengkap dengan atribut dan methodnya. Semua identifier harus ada (tidak harus dalam 1 class., boleh tersebar). Contoh method lengkap yang menggunakan parameter input, tanpa parameter, dengan return value, tanpa return value, serta kombinasinya! Kemudian implementasikan menjadi kode program!

1. Class Diagram : Date

Date
+ day : int + month : int + year : int
+ setDay(day : int) + getDay() : int + setMonth(month : int) + getMonth() : int + setYear(year : int) + getYear() : int

Listing program

```

public class Date{
    public int day;
    public int month;
    public int year;

    public void setDay(int day){
        this.day = day;
    }

    public int getDay(){
        return day;
    }

    public void setMonth(int month){
        this.month = month;
    }
}

```

```

        public int getMonth(){
            return month;
        }

        public void setYear(int year){
            this.year = year;
        }

        public int getYear(){
            return year;
        }
    }

```

2. Class Diagram : Luas

Luas
+ int : panjang + int : lebar + double : luas
+ setPanjang(panjang : int) + getPanjang() : int + setLebar(lebar : int) + getLebar() : int + setLuas(panjang : int, luas : int) + getLuas() : int

Listing Program

```

public class Luas{
    public int panjang;
    public int lebar;
    public double luas;

    public void setPanjang(int panjang){
        this.panjang = panjang;
    }

    public int getPanjang(){
        return panjang;
    }

    public void setLebar(int lebar){
        this.lebar = lebar;
    }

    public int getLebar(){
        return lebar;
    }

    public void setLuas(int panjang, int lebar){
        luas = (double) (panjang * lebar)
    }

    public double getLuas(){
        return luas;
    }
}

```

```
    }  
}
```

3. Class Diagram : Mahasiswa

Mahasiswa
+ nrp : String
+ Mahasiswa (i : int)
+ setNrp(i : int)
+ getNrp(i : int)

Listing Program

```
public class Mahasiswa{  
    public String nrp;  
  
    public Mahasiswa(int i){  
        nrp = i;  
    }  
  
    public void setNrp(int i){  
        nrp = i;  
    }  
  
    public int getNrp(){  
        return nrp;  
    }  
}
```